数理解析研究所講究録
90巻 1970年 143-161

A Method for Optimizing Control of Multimodal Systems using Fuzzy Automata

Kiyoji Asai
University of Osaka Prefecture

Seizo Kitajima
Osaka City University

SUMMARY

This paper deals with a new method for optimizing control of multimodal systems by using fuzzy automata.

The concept of fuzzy sets was introduced by L.A.Zadeh for the purpose of defining a fuzzy relation between elements and their set. One of the basic points of this concept is the idea of the membership function by which the grade of a fuzzy relation can be represented as the real positive numbers of the closed interval [0,1]. If A is a fuzzy set with a generic element x, then the nearer the value of the membership function $f_A(x)$ approaches to unity, the higher the grade of membership of x in A will grow.

W.G.Wee and K.S.Fu have proposed a fuzzy automaton, in which the transition from a state to another state may be executed on the basis of the membership function between these two states, and shown a learning system using this automaton.

The authors have formulated a kind of fuzzy automata which have many outputs in a branch or a state, and have shown the learning behavior of new optimizing control systems using these automata. A part of these works have been presented in preliminary report.

The domain of the objective function for optimization is divided into some number of sub-domains, and every sub-domain is also divided into some unit domains corresponding to the outputs from a branch or a state. Therefore, a global search can be executed by deciding the optimum output after the search of the sub-domain in which the optimum output may be probably included.

The membership functions of the automata can be adjusted on the basis of the objective function, and the operation of self-

organization in the automata is more clearly performed than in case of the optimizing control systems using the stochastic automata reported by G.J.McMurtry and K.S.Fu.

A Method for Optimizing Control of Multimodal Systems using Fuzzy Automata

Kiyoji Asai
University of Osaka Prefecture
Osaka, Japan

Seizo Kitajima
Osaka City University
Osaka, Japan

ABSTRACT

This paper deals with a new method for optimizing control of multimodal systems by using fuzzy automata.

Two classes of fuzzy automata corresponding to Mealy and Moore type ordinary automata are formulated, in which the transition may be executed on the basis of the membership function between these two states, and these automata are used as the optimizing controller for a global search of the optimum of a multimodal systems.

The ideas of the higher-order transition in the automata and the partition of the domain of objective function are introduced to the method of optimizing control, and the control systems can hold the true optimum at small hunting loss without keeping any local optimum. The operation of self-organization is performed to optimize the system in the automata.

INTRODUCTION

The concept of fuzzy sets was introduced by L.A.Zadeh[1] for the purpose of defining a fuzzy relation between elements and their set. One of the basic points of this concept is the idea of the membership function by which the grade of a fuzzy relation can be represented as the real positive numbers of the closed interval $[0,1]$. If A is a fuzzy set with a generic element x, then the nearer the value of the membership function $f_A(x)$ approaches to unity, the higher the grade of membership of x in A will grow.

W.G.Wee and K.S.Fu [2] have proposed a fuzzy automaton, in which the transition from a state to another state may be executed on the basis of the membership function between these two states, and shown a learning system using this automaton. The authors have formulated a kind of fuzzy automata which have many outputs in a branch or a state, and have shown the learning behavior of new optimizing control systems using these automata. A part of these works have been presented in preliminary report(3).

The domain of the objective function for optimization is divided into some number of sub-domains, and every sub-domain is also divided into some unit domains corresponding to the outputs from a branch or a state. Therefore, a global search can be executed by deciding the optimum output after the search of the sub-domain in which the optimum output may be probably included.

The membership functions of the automata can be adjusted on the basis of the objective function, and the operation of self-organization in the automata is more clearly performed than in case of the optimizing control systems using the stochastic automata reported by G.J.McMurtry and K.S.Fu(4).

FUZZY AUTOMATA FOR OPTIMIZING CONTROL SYSTEMS

A fuzzy automaton is a kind of automaton which will transite from a state to another state or the same state via the branch whose membership function is the largest one among those of all branches diverging from the state, when an input is applied.

Two classes of fuzzy automata similar to the ordinary automata may be defined as follows.

(i) Mealy type fuzzy automata : The output will be sent out when a transition is executed between two states via a branch.

(ii) Moore type fuzzy automata : The output will be sent out when a transition has arrived at the next state from a state.

(a) Mealy type fuzzy automata

The Mealy type fuzzy automata may be expressed as shown in (1).

$$M = \left\{ S, X, U, F(u/x) \right\} \tag{1}$$

where $S = \{ s_1, s_2, ----, s_\nu \}$: set of ν states

$X = \{ x_1, x_2, ----, x_\mu \}$: set of μ inputs

$U = \{ u_1, u_2, ----, u_\xi \}$: set of ξ outputs

$F(u/x) = \nu \times \nu$ fuzzy transition matrix.

In case of single input, the fuzzy transition matrices may be shown in (2).

$$F(u_h/x_1) = \left[f_{ij}(u_h/x_1) \right] \tag{2}$$

where $h = 1, 2, ---, \xi$

$f_{ij}(u_h/x_1)$: membership function that the automaton will go to state s_j from state s_i and send out the output u_h when the input x_1 is applied.

$i, j = 1, 2, ---, \nu$

The membership function for the transition by a branch leading from a state s_i to a state s_j is shown as

$$f^1_{ij}(u/x_1) = \max\left\{ f_{ij}(u_1/x_1),\ f_{ij}(u_2/x_1),\ --- \right.$$
$$\left. ,\ f_{ij}(u_\xi/x_1) \right\} \tag{3}$$

By using the symbols of algebraic sum instead of the symbol max, (3) can be rewritten as

$$f^1_{ij}(u/x_1) = f_{ij}(u_1/x_1) + f_{ij}(u_2/x_1) + ---$$
$$+ f_{ij}(u_\xi/x_1) \tag{4}$$

So the first order fuzzy transition matrix will be

$$F^1(u/x_1) = F(u_1/x_1) + F(u_2/x_1) + --- + F(u_\xi/x_1) \quad (5)$$

Then, the membership function for a path in which the second order transition may be executed from a state s_i to a state s_j via a state s_k is given by

$$f^2_{ij}(u/x_1) = \max_k \left\{ \max_{h_1} \max_{h_2} \left[\min(f_{ik}(u_{h_1}/x_1), f_{kj}(u_{h_2}/x_1)) \right] \right\}$$

$$= \max_k \left\{ \min \left[\max_{h_1} f_{ik}(u_{h_1}/x_1), \max_{h_2} f_{kj}(u_{h_2}/x_1) \right] \right\} \quad (6)$$

Using (3), we have

$$f^2_{ij}(u/x_1) = \max_k \left\{ \min \left[f^1_{ik}(u/x_1), f^1_{kj}(u/x_1) \right] \right\} \quad (7)$$

By using the symbols of algebraic sum and product instead of the symbols max and min respectively, (7) can be rewritten as

$$f^2_{ij}(u/x_1) = f^1_{i1}(u/x_1) \cdot f^1_{1j}(u/x_1) + f^1_{i2}(u/x_1) \cdot f^1_{2j}(u/x_1) + --- + f^1_{i\nu}(u/x_1) \cdot f^1_{\nu j}(u/x_1) \quad (8)$$

So the second order fuzzy transition matrix will be

$$F^2(u/x_1) = F^1(u/x_1) \cdot F^1(u/x_1) \quad (9)$$

By the same manner, the k th-order fuzzy transition matrix will be obtained as

$$F^k(u/x_1) = F^1(u/x_1) \cdot F^1(u/x_1) \cdot --- \cdot F^1(u/x_1) \quad (10)$$

Thus the transition matrix of the Mealy type fuzzy **automaton** may be given as the sum of the fuzzy transition matrices when the first order transition is executed, and as the product of the first order fuzzy transition matrices when the higher-order transition is executed.

In case of multi-input sequence, the membership function for a path in which the k th-order transition may be executed from state s_i to state s_j via serial branches is given as follows.

$$f^k_{ij} = f^k_{ij}\left\{ s_i,\ x^k,\ s_j \right\}$$

$$= \sup_{s_p, s_q, ---, s_v} \min \left[f_{ip}\left\{ s_i, x(1), s_p \right\},\ f_{pq}\left\{ s_p, x(2), s_q \right\},---,\ f_{vj}\left\{ s_v, x(k), s_j \right\} \right] \tag{11}$$

$$= \sup_{s_p, s_q, ---, s_v} \min \left[f^1_{ip}(u/x_1), f^1_{pq}(u/x_2),\ ---,\ f^1_{vj}(u/x_k) \right] \tag{12}$$

(b) Moore type fuzzy automata

The Moore type fuzzy automata may be expressed as shown in (13).

$$M' = \left\{ S, X, U, F(x), G(s,u) \right\} \tag{13}$$

where $F(x)$: $\nu \times \nu$ transition matrix

$G(s,u)$: $\nu \times \xi$ output matrix

The transition matrix may be shown in (14).

$$F(x_1) = \left[f_{ij}(x_1) \right] \tag{14}$$

where $1 = 1,2,---,\nu$

$f_{ij}(x_1)$: membership function for the transition from state s_i to state s_j when an input x_1 is applied.

The output matrix may be shown in (15).

$$G(s,u) = \left[g_{jh}(u_h)\right] \tag{15}$$

where $g_{jh}(u_h)$: membership function for the choice of an output u_h at the state s_j when the transition has been executed to the state s_j.

In the Moore type, two methods of transition may be defined as follows.

(i) The transition between two states may be executed by using the transition matrix, and a proper output may be selected by using the output matrix at the state to which the transition has been executed.

(ii) Both transition between two states and choice of output may be executed on the basis of the matrix of the product of the transition matrix by the output matrix.

In the case (i), the k th-order transition matrix may be given by

$$F^k(x_1) = F(x_1)\cdot F(x_1)\cdot \text{----} \cdot F(x_1) \tag{16}$$

In case of the multi-input sequence, the membership function for a path in which the k th-order transition may be executed from state s_i to state s_j via serial branches is given as follows.

$$f^k_{ij} = \sup_{s_p, s_q, --, s_v} \min\left[f^1_{ip}(x_1), f^1_{pq}(x_2), --, f^1_{vj}(x_k)\right] \tag{17}$$

Then an output may be selected by the output matrix.

In the case (ii), the membership function for the transition from state s_i to state s_j when a single input x_1 is applied may be given as

$$f^1_{ij} = \max_h\left\{\min(f_{ij}(x_1),\ g_{jh}(u_h))\right\}, \tag{18}$$

and the transition matrix whose entries are given as (18) may be shown by

$$F^1(u/x_1) = \left[f^1_{ij} \right] \tag{19}$$

Then, the membership function for a path in which the second order transition may be executed from a state s_i to a state s_j via a state s_k is given as follows.

$$f^2_{ij} = \max_k \left\{ \max_{h_2} \max_{h_1} \left[\min(f_{ik}(x_1), g_{kh_1}(u_{h_1}), f_{kj}(x_1), g_{jh_2}(u_{h_2})) \right] \right\} \tag{20}$$

$$= \max_k \left\{ \min(f^1_{ik}, f^1_{kj}) \right\} \tag{21}$$

So the second order transition matrix will be

$$F^2(u/x_1) = F^1(u/x_1) . F^1(u/x_1) \tag{22}$$

By the same manner, the k th-order transition matrix $F^k(u/x_1)$ will be given by the k th power of the first order transition matrix $F^1(u/x_1)$.

Thus the transition between two states and the choice of output of the Moore type fuzzy automaton may be executed by using (19) for the first order transition or the product of the first order transition matrices for higher-order transition. In this case the fuzzy automaton will go to state s_j from state s_i and send out of the selected outputs when an input sequence is applied. This behavior is similar to that of the Mealy type fuzzy automaton.

So the transformable characteristics between these two automata will be shown as follows.

Assuming that f'_{ij} is the membership function for the transition from state s_i to state s_j and g'_{jh} is the membership function for the choice of output u_h at the state s_j, the transition matrix of the Mealy type fuzzy automaton shown in (2) may be written as follows.

$$F(u_h/x_1) = \left[f'_{ij}\right] \cdot \mathrm{dig}\left[g'_{jh}\right] \tag{23}$$

where dig $\left[g'_{jh}\right]$: diagonal matrix $\left[g'_{1h}, g'_{2h}, ---, g'_{\nu h}\right]$

So (4) will be rewritten as

$$f'_{ij}(u/x_1) = f'_{ij} \cdot g'_{j1} + f'_{ij} \cdot g'_{j2} + --- + f'_{ij} \cdot g'_{j\xi} \ , \tag{24}$$

and

$$f'_{ij}(u/x_1) = \max_h \left\{ \min(f'_{ij}, g'_{jh}) \right\} \tag{25}$$

If

$$f'_{ij} = f_{ij}(x_1) \quad , \quad g'_{jh} = g_{jh}(u_h) \tag{26}$$

then (25) is equal to (18).

The first transition matrix $F^1(u/x_1)$ whose entries are given by (24) is equal to $F^1(u/x_1)$ of (5) if (26) is satisfied. The higher-order transition matrices of both Mealy and Moore type fuzzy automata may be expressed by the product of the first order transition matrix. If (23) and (26) are satisfied, then next equations are obtained.

$$f_{ij}(u_1/x_1) = f_{ij}(x_1) \cdot g_{j1}(u_1) \quad , \quad f_{ij}(u_2/x_1) = f_{ij}(x_1) \cdot$$

$$g_{j2}(u_2) \quad , \quad --- \quad , \quad f_{ij}(u_\xi/x_1) = f_{ij}(x_1) \cdot g_{j\xi}(u_\xi)$$

Since the transition matrix of the Mealy type fuzzy automaton will equal to that of the Moore type, the Mealy type fuzzy automaton may be transfered to the Moore type and conversely.

The fuzzy automata described above may be applied to an optimizing control system with the controlled system whose characteristics are unknown. In this case, on the basis of the objective function, the membership functions which are entries of the transition matrix may be corrected by a learning operation and the optimum may be searched.

Either Mealy or Moore type fuzzy automaton may be useful for the optimizing control, but if it is assumed that the input is single and all outputs at each state are different so as to simplify the structure of automaton, the number of outputs at each state may be ξ/ν. If $\xi = \nu \times \nu$ then ν outputs at each state or an output at a branch may be prepared in the Moore type or the Mealy type fuzzy automaton respectively.

The following paragraphs deal with the case that the single input fuzzy automaton with $(\nu \times \nu)$ outputs will be applied to the optimizing control system.

OPTIMIZING CONTROL SYSTEMS

The block diagram of an optimizing control system using the fuzzy automaton is shown in Fig. 1. The outline of performance of the system is as follows.

(i) When the n th input $x(n)$ arrives at the fuzzy automaton, a transition will be executed from state s_i to state s_j on the basis of the membership function $f_{ij}(n)$, and the n th output $\mathcal{U}(n)$ which is generally called "control variable" will be sent out.

(ii) The n th output $\mathcal{Y}(n)$ will be sent out from the controlled system whose input-output characteristic is unknown, corresponding to the control variable $\mathcal{U}(n)$.

(iii) The n th objective function will be calculated from $\mathcal{U}(n)$ and $\mathcal{Y}(n)$ in the computer for objective function.

(iv) The n+1 th membership function $f_{ij}(n+1)$ will be determinded by modifing $f_{ij}(n)$ on the basis of the result wheather the n th objective function approaches toward the optimum value, that is success, or not, that is failure. By increasing or decreasing the membership function of the branch, the grade of transition by the branch of success or failure will be raised or made low respectively.

For simplicity of explanation, the system performance has been mentioned above on the case of optimizing control by the first order transition in the automaton, without considering the sub-domains, but the authors have introduced the ideas of the higher-order transition and the partition of the domain of objective function in order to realize a global search and the control at small hunting loss, and so its details will be given below.

(i) In the first, the membership functions f_{ij} and g_{jh} are set in the middle value 0.5 between 0 and 1, so that transition from a state to an arbitrary state can be executed.

(ii) For the purpose of finding a path that loops from a state s_i to the same state s_i via K branches, a K th-order transition matrix is calculated by (10), and the maximum value $\left[f^K_{ii,max}\right]$ of membership function on the diagonal entries in the matrix and the number $[M]$ of these maximum membership functions are found.

(iii) Comparing M with K, if **M<K,** M th-order transition matrix is calculated in order to remove overlapping branches, and then the same procedure as (ii) is executed. And, this procedure is repeated until all overlapping branches are removed from the loop path. Let the maximum value of membership function and the number of the maximum membership functions thus obtained be $f^{k}_{ii,max}$ and m respectively.

(iv) Since the k branches on the single loop leading from s_i to s_i should have the membership functions that are equal to or larger than $f^{k}_{ii,max}$, the loop path may be found by comparing $f^{k}_{ii,max}$ with the membership functions for k branches leading from s_i to s_i. In this case, if the single loop leading from s_i to s_i can not be found, the procedures (ii) - (iv) should be repeated for the k -1 th-order transition matrix or the transition matrix of lower order than the k -1 th until the single loop is obtained.

(v) When the transition is executed on the path obtained in (iv), k control variables are sent from the automaton to the controlled system from which k outputs will be sent out, and then k objective functions $I_k(n)$ and the mean value $\bar{I}(n)$ of all objective functions thus far obtained are calculated.

(vi) comparing k objective functions $I_k(n)$ with the mean value $\bar{I}(n)$, the n+1 th membership functions $f_{ij,k}(n+1)$ and $g_{jh,k}(n+1)$ are respectively determined from the n th membership functions $f_{ij,k}(n)$ and $g_{jh,k}(n)$ by using the algorithm as shown in (27).

$$\left.\begin{aligned} & f_{ij,k}(n+1) = \alpha f_{ij,k}(n) + (1-\alpha) \\ & \text{if } I > \bar{I} \qquad \text{(success)} \\ & g_{jh,k}(n+1) = \alpha g_{jh,k}(n) \\ & \text{if } I \leq \bar{I} \qquad \text{(failure)} \end{aligned}\right\} \tag{27}$$

where $\alpha = 1- \left|(I-\bar{I})/\bar{I}\right|$, but we regard α as 0.5 if the calculated value of α is equal to or less than 0.5, and also regard α as 0.99 if the calculated value of α is equal to 1.

(vii) Since the optimum output may be probably included to the state s_j, if the membership functions f_{ij} modified by (27) are larger than some value (e.g.0.8) in case of **success**, the outputs whose membership functions g_{jh} are larger than some value (e.g.0.45) will be applied to the controlled system, and then the membership functions of the branches that have not been modified will be set to the maximum value of g_{jh} in order to facilitate to go to the state s_j. The membership function f_{jj} will be magnified if f_{ij} glows very large (e.g.0.9).

(viii) Since the optimum output may not be probably included to the state s_j if $g_{jh}(n+1)$ glows less than some value (e.g.0.2), f_{ij} will be made small.

From the explanation mentioned above, it will be seen that the operation of self-organization has been performed to optimize the system in the automaton as follows.

The membership functions of the branches that contribute to success will approach to 1, while those of the branches that cause failure will approach to 0. Therefore, the proper ones for getting the optimum are selected among all the branches in the automaton while the unsuitable ones are weeded out.

SIMULATION RESULTS

A simulation study has been carried out for the purpose of investigating the behaviors of the optimizing control using the system descrived above.

In this computer simulation, the following equation(5) has been used as an objective function that includes the characteristics of controlled system.

$$I(u_1,u_2) = (1 + 8u_1 - 7u_1^2 + 7/3u_1^3 - 1/4u_1^4)u_2^2 \cdot e^{-u_2} \qquad (28)$$

This objective function is the function with two variables u_1,u_2 as control variables and includes an optimum point, a local optimum point and a saddle point. we assume that the measured value of $I(u_1,u_2)$ includes an observational error in proportion to random number with normal distribution (standard deviation

σ) in the calculated value of (28).

Examples of the results of simulation study are shown in Fig. 2 and Fig. 3.

Fig.2 is the simulation result of the case of the Moore type fuzzy automaton. From Fig.2, it may be seen that an optimizing control has been executed as follows.

A global search was executed at early stage of learning, but the operation of reducing the domain of search was executed with the progress of the learning.

The states (sub-domains) ③ , ④ , ⑦ were searched over all outputs (unit-domains) since the optimum may be probably included in some domain among these three domains, and then a loop path which connects the states ③ , ④ , ⑦ were formed. At the final stage of learning, the state ④ was selected and the optimum was maintained by the transition by the self loop at the state ④ .

Fig.3 is the simulation result of the case of the Mealy type.

From Fig.3, it may be seen that an optimizing control has been executed as follows.

At first the trials were executed at 7 points over a wide range of values of control variables as shown by the dot • in Fig.3, and then the learning was begun. At an early stage of learning, the learning characteristics may be similar to that of the Moore type, but a loop path connecting the specific states will be formed with the progress of the learning. At the final stage of learning, the transition was repeatedly executed between the state corresponding to the optimum and the state corresponding to a quasi optimum.

CONCLUSIONS

In the preceding paragraphs, the formulation of the fuzzy automata and the optimizing control systems using these automata has been described.

From the results of theoretical and experimental investigation, it has been concluded that every Mealy type fuzzy automaton can transfer to a Moore type fuzzy automaton and conversely if these both automata have same number of states, and that the optimizing control systems using these automata are able to avoid the continuation of trials in the vicinity of a local optimum by global search and to get the performance of small hunting loss at the steady state.

These merits are due to the ideas of the higher-order transition and the partition of the domain of objective function.

It has been also seen that the operation of self-organization is performed to optimize the system in the automata.

The authors would like to express their appreciation to prof. L.A.Zadeh and prof. K.S.Fu for valuable discussions.

REFERENCES

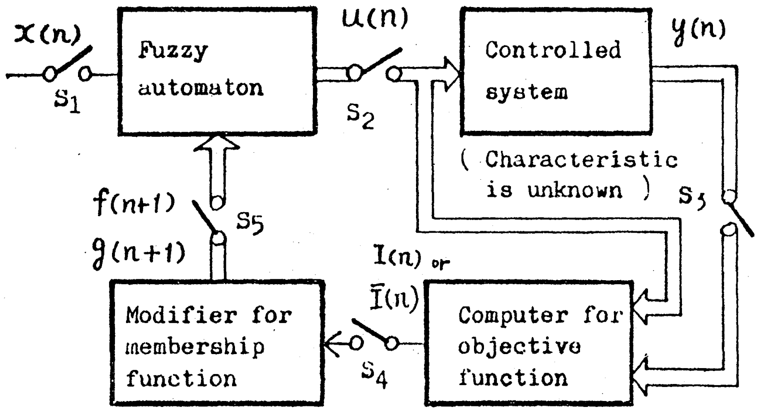
(1) Zadeh, L.A., "Fuzzy Sets", INFORMATION AND CONTROL, Vol.8, No. 3, June, 1965.

(2) Wee, W.G. & Fu, K.S., " A Formulation of Fuzzy Automata and its Application as a Model of Learning Systems", IEEE TRANS., Vol. SSC-5, No. 3, July, 1969.

(3) Hirai, H., Asai, K. & Kitajima, S., "Fuzzy Automaton and its Application to Learning Control Systems", MEMOIRS OF THE FACULTY OF ENG., OSAKA CITY UNIV., Vol. 10, Dec., 1968.

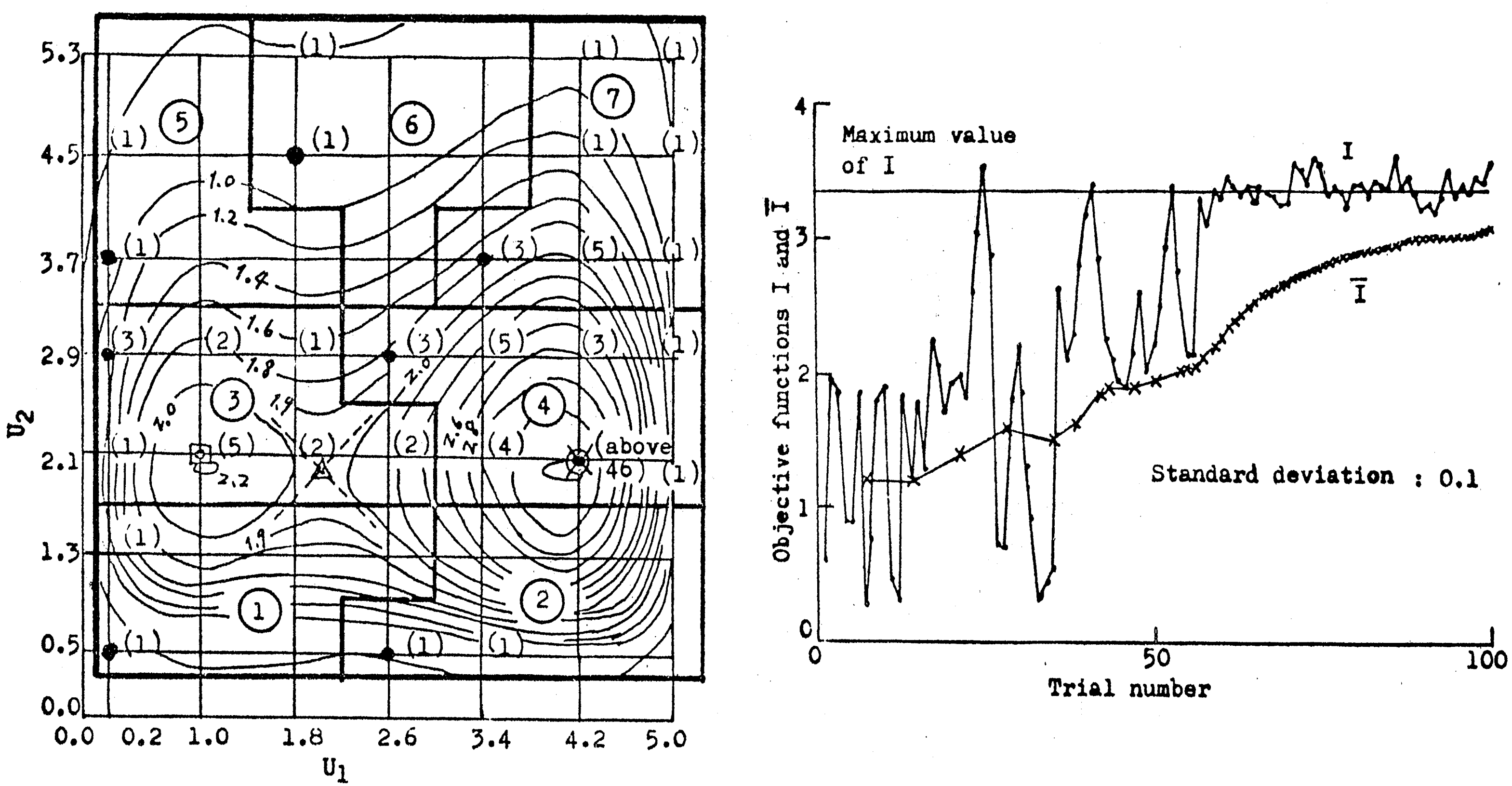
(4) McMurtry, G.J. & Fu, K.S., "A Variable Structure Automaton used as a Multimodal Searching Technique", IEEE TRANS., Vol. AC-11, No. 3, July, 1966.

(5) Hill, J.C., "A Hillclimbing Technique using Piecewise Cubic Approximation", PH. D. DISSERTATION, PURDUE UNIV., 1964.



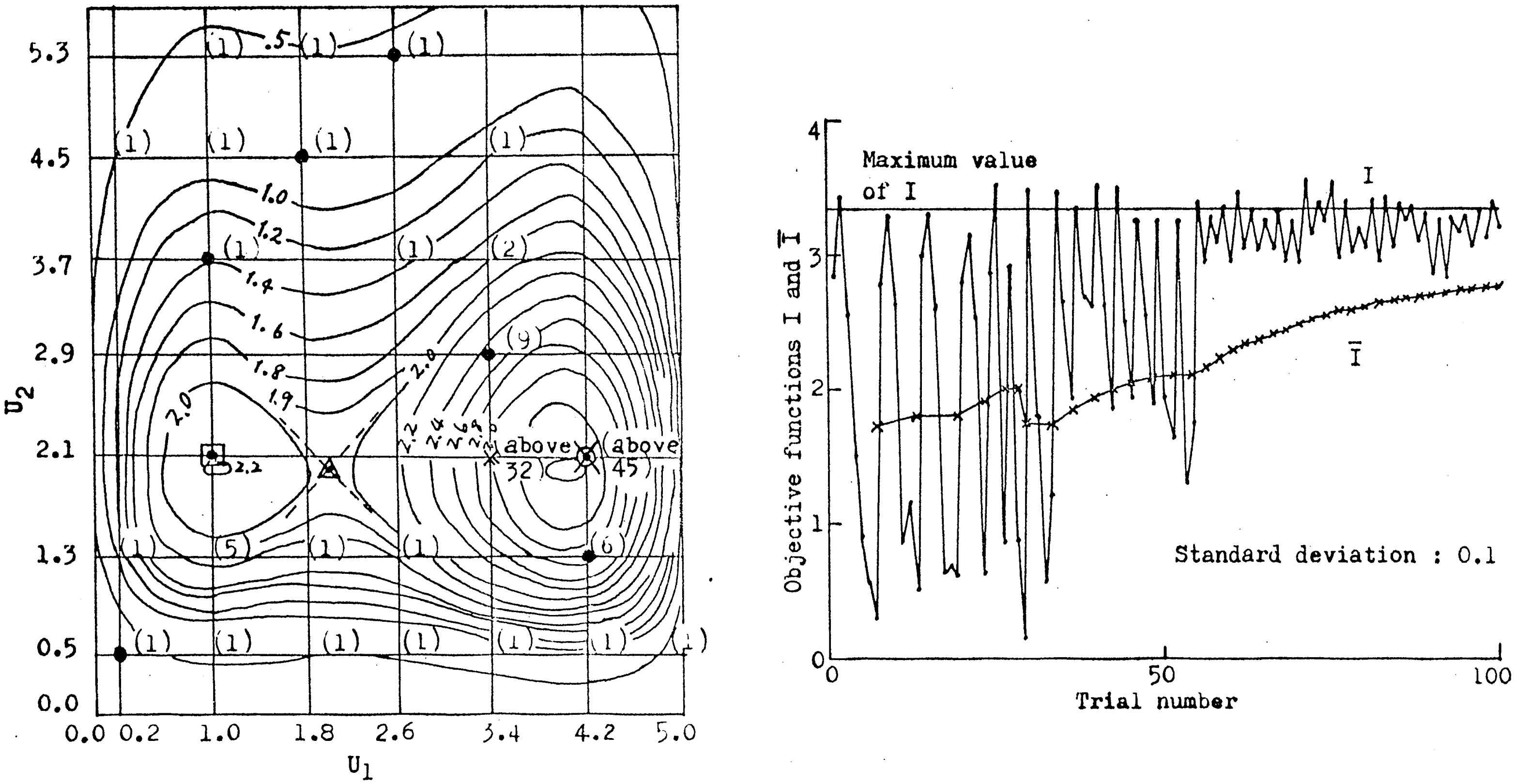
S_1, S_2, S_3, S_4, S_5 : Samplers

Fig. 1 Block diagram of optimizing control system.



• : Trial points at starting time(7 points),
×: Trial points at final stage(1 point),
⊙: Optimum point, ⊡: Local optimum point,
△: Saddle point, ( ) : Number of trials,
○: State number

Fig. 2 An example of results of simulation study (case of Moore type).



●: Trial points at starting time(7 points),
×: Trial points at final stage(2 points),
⊙: Optimum point, ⊡ : Local optimum point,
△: Saddle point, ( ) : Number of trials

Fig. 3 An example of results of simulation study (case of Mealy type).